

DAQ Status

John Haggerty
Brookhaven National Laboratory

February 11, 2004



Systems Status

- **GL1** Occasionally the GL1DCM scrambles the GL1 data, but it's quite rare and the cure is simple
- **DCM's** Hardly any problems this year; maybe once in a TEC crate
- **GTM's** We even have some spares; Steve Boose finished a putative PPG replacement
- **FEM's** Well-behaved, except for 1 DC.E, 2 EMC.E.B, some missing TEC
- **ARCNET** After an early failure in a hub card, I've forgotten about it
- **JSEB** Still a nagging problem requiring padding banks; but then I couldn't see it when I looked for it, and then I got sidetracked
- **OPC** Clients and servers have been stable; logging continuously for a month
- **HV** Hardware pretty stable; reasonable stock of spares; Martin's new server faster and more stable

Software Status

- **PartitionServer** What's that do, anyway?
- **level1gui** Faster and less prone to hang with PostgreSQL
- **EvBServer** Initially seemed to have lots of problems, but after switch firmware was upgraded and EvB components stopped vanishing from the network, it has been much more stable; still, there are crashes that should be debugged and a number of improvements to make (callbacks, better treatment of multiple buffer boxes)
- **aml** Pretty stable when running, but there are some peculiarities when starting up occasionally
- **rc** Occasional hangs, some peculiarities (always download before starting GL1 rc?), but faster without multiple initializations
- **arcnet** OK, I guess; haven't looked lately
- **Monitoring** I really like the web plots; ran pretty smoothly unattended through QM

Multievent Buffering Status

- The good news—the Pad Chamber runs better with the new firmware than the old; no funny business with the beam clock counter, and no problem that we've seen with multievent buffering
- More good news—the whole detector runs ok with multievent buffering turned on
- Yet more good news—the livetime is much better with multievent buffering; maybe not 99%, but above 90% above 1 kHz

The Bad News About Multievent Buffering

- Although multievent buffering works in the MUTR, after a short time, the data are scrambled; the AMU cells reported are duplicated (usually 3 of the 4 samples)
- We're studying the problem (Ming, Jack, MUTR experts, and me)
- So far, we have not reproduced the problem with Forced Accepts, but we haven't done a test in which all amu cells are used, so it's a case of "absence of evidence not being evidence of absence"
- We've tinkered with the conversion time and the endat time, but we have not seen evidence in the design that there are any "magic" settings
- There could be "magic" trigger separations that cause this but we haven't found them, either
- It's possible that the conversion time is *much* longer than we think it should be in some situations
- We need some more testing (run MUTR in test mode, random triggers in standalone)
- We may need to revise the FPGA code in the MUTR to fix this; it can be done by ARCNET, although it is a somewhat frightening process
- If we need new FPGA code, we're going to have to be very sure that it really, really works before we flash 500 FEM's
- If MUTR gets solved, we have developed another problem in which the GL1 gets read errors after a few thousand events; it could be a problem with the GL1, LL1's , or GL1DCM... still collecting evidence

Summary

- We've been running pretty well the past few weeks after a somewhat rocky start; we've come infinitely far since December
- My ersatz spreadsheet says that we have over 500M events recorded
- Compression is working and has not caused us problems thanks to Martin and Jinagyong (we are reading this data, right?)
- We've logged data at a rate of 330 Mbyte/sec (uncompressed)
- The average livetime in a fill isn't too bad (75%?)
- Database swap, OS upgrades, new CORBA version haven't caused much problem
- Still some (dim) hope for multievent buffering this run; if we need new FPGA code, we might want to risk flashing it before the pp run
- Still a variety of bugs, hangs, crashes, and mysteries that deserve investigation
- Still some desire to push ahead with EMCAL firmware